Journal of Computational and Applied Mathematics 33 (1990) 133–140
North-Holland

Spline product quadrature rules for Cauchy singular integrals *

C. DAGNINO and E. SANTI
Facoltà di Ingegneria, Università degli Studi di L'Aquila, Località Monteluco, 67040 Roio Poggio, L'Aquila, Italy

Received 30 November 1989
Revised 7 June 1990

Abstract: In this paper product quadrature rules, based on cubic spline interpolation, are obtained for the numerical evaluation of Cauchy singular integrals and an error bound is proposed. Some numerical examples that test the performance of the rules are given.

Keywords: Cauchy singular integrals, cubic B-splines.

1. Introduction

For $\lambda \in (-1, 1)$ let $I(uf; \lambda)$ denote the Cauchy principal value integral

$$I(uf; \lambda) = \int_{-1}^{1} u(x) \frac{f(x)}{x-\lambda} \, dx$$
$$= \lim_{\epsilon \to 0} \left\{ \int_{-1}^{\lambda-\epsilon} + \int_{\lambda+\epsilon}^{1} \right\} u(x) \frac{f(x)}{x-\lambda} \, dx, \tag{1}$$

where $u(x)$ is a nonnegative weight function on $(-1, 1)$ such that $\int_{-1}^{1} u(x)/(x-\lambda) \, dx$ exists.

A practical method for evaluating (1) is a product quadrature rule of the form

$$I(uf; \lambda) \approx Q_n(uf; \lambda) = \sum_{i=0}^{n+1} w_i(\lambda) f(x_i), \tag{2}$$

where the points $\{x_i\}$ are given on a mesh defined by $-1 \leqslant x_0 < x_1 < \cdots < x_n < x_{n+1} \leqslant 1$ and the weights $\{w_i(\lambda)\}$ are chosen so that (2) is exact when f belongs to a family of linearly independent functions $F_n = \{\Phi_j(x)\}_{j=0}^{n+1}$.

In this case the weights $\{w_i(\lambda)\}$ satisfy the linear system

$$\sum_{i=0}^{n+1} w_i(\lambda) \Phi_j(x_i) = I(u\Phi_j; \lambda), \quad j = 0, \ldots, n+1. \tag{3}$$

A requirement in (3) is that the matrix $\{\Phi_j(x_i)\}$ be nonsingular.

* Work sponsored by the Ministero della Pubblica Istruzione of Italy.

Product rules of the type (2) can have a useful application in the numerical treatment of the Cauchy singular integral equation

$$a(\lambda)u(\lambda)f(\lambda)+\frac{b(\lambda)}{\pi}I(uf;\ \lambda)=g(\lambda),$$

where $a(\lambda)$, $b(\lambda)$, $g(\lambda)$ are given input functions, $f(x)$ is the unknown function and the weight function $u(x)$ generally exhibits square root singularities at the end points [2,17,23, etc.].

It is to be recalled that quadrature rules (2), based on global polynomial interpolation at zeros of orthogonal polynomials or practical abscissas, converge very fast for some suitable classes of functions f [4–6,10,15,16,21,24,25] and there is an explicit formula for their weights [4,14,16,24,25].

However, in some practical applications of rules (2) one cannot always place the abscissas at zeros of orthogonal polynomials. In contrast, local methods, based on piecewise polynomial approximations, can afford a flexible choice of the node points.

Although progress has been made on various aspects of product quadratures for Cauchy singular integrals [22], very little attention has been paid to local methods mainly based on piecewise polynomial approximations [17,18,23].

The present paper is devoted to an investigation of quadratures (2) based on cubic splines. In this case the functions $\{\Phi_j\}$ are chosen to be B-splines [11,12] and the linear system (3) gives rise to a totally positive, almost tridiagonal matrix. Hence, (3) can be stably solved by Gaussian elimination without partial pivoting [13].

In Section 2 spline product rules for (1) are presented and a bound for the quadrature error is proposed.

In Section 3 we describe a computational procedure for generating the above rules and in Section 4 we give some numerical examples.

Finally in Section 5 we make some concluding remarks.

2. Spline product quadrature rules

Let n knots $-1\leqslant t_1<t_2<\cdots<t_n\leqslant 1$ and $(n+2)$ abscissas of integration $-1\leqslant x_0<x_1<\cdots<x_n<x_{n+1}\leqslant 1$ be given. The product integration rule $Q_n(uf;\ \lambda)$ is defined by (2) and the weights $\{w_i(\lambda)\}$ are chosen so that

$$Q_n(uf;\ \lambda)=\int_{-1}^{1}u(x)\frac{s(x)}{x-\lambda}\,\mathrm{d}x, \tag{4}$$

where $s(x)$ is a cubic spline with knots $\{t_i\}_{i=1}^{n}$ [19]. That is, $s''\in C[-1,1]$ and $s(x)$ is a polynomial of degree at most three in the interval $(t_i,\ t_{i+1})$, $i=1,\dots,n-1$. As remarked in [1], if one adds knots $t_{-2}\leqslant t_{-1}\leqslant t_0\leqslant -1$ and $1\leqslant t_{n+1}\leqslant t_{n+2}\leqslant t_{n+3}$ one can introduce a cubic B-spline basis $\{B_{n,j}^{(4)}\}$ [11] for the cubic splines with knots $\{t_j\}_{j=-2}^{n+3}$ and rewrite (2) and (4) in the form

$$\sum_{i=0}^{n+1}B_{n,j}^{(4)}(x_i)w_i(\lambda)=I\big(uB_{n,j}^{(4)};\ \lambda\big),\quad j=2,\dots,n+3. \tag{5}$$

The computational aspects for constructing the matrix with elements $\{B_{n,j}^{(4)}(x_i)\}$ and the vector with elements $\{I(uB_{n,j}^{(4)};\ \lambda)\}$ are deferred until the next section.

From (5) and the Schoenberg–Whitney Theorem [11] one obtains a necessary and sufficient condition for the rules to exist, namely

$$t_{i-2} < x_i < t_{i+2}, \quad i = 0, \dots, n+1. \tag{6}$$

For our purposes, once the knots are fixed, the following choice of abscissas of integration is used:

$$\begin{aligned} &x_0 = t_1, \qquad x_1 = \tfrac{1}{2}(t_1 + t_2), \qquad x_j = t_j, \quad j = 2, \dots, n-1, \\ &x_n = \tfrac{1}{2}(t_n + t_{n-1}), \qquad x_{n+1} = t_n. \end{aligned} \tag{7}$$

This choice of collocation points ensures that (6) holds and that all abscissas of integration lie in $[-1, 1]$.

We remark that a useful representation for $Q_n(uf;\ \lambda)$ is the following

$$Q_n(uf;\ \lambda) = \int_{-1}^{1} u(x) \frac{\varphi^{(n)}(x)}{x - \lambda}\, \mathrm{d}x, \tag{8}$$

where $\varphi^{(n)}$ is the unique cubic spline interpolating f at the points $\{x_i\}_{i=0}^{n+1}$ and satisfying the "not-a-knot" end condition [11].

From (8), the weights $\{w_i(\lambda)\}$ may also be expressed in terms of the cardinal splines $\varphi_i(x)$, $i = 0, \dots, n+1$, which satisfy

$$\varphi_i(x_j) = \delta_{ij}, \quad i,\ j = 0, \dots, n+1,$$

by

$$w_i(\lambda) = \int_{-1}^{1} u(x) \frac{\varphi_i(x)}{x - \lambda}\, \mathrm{d}x, \quad i = 0, \dots, n+1. \tag{9}$$

It is to be noted that the above rules are an extension to the numerical evaluation of Cauchy singular integrals, of standard product rules recently proposed in [1] and also investigated in [7,8].

Now we shall consider the quadrature error

$$R_n(uf;\ \lambda) = I(uf;\ \lambda) - Q_n(uf;\ \lambda) \tag{10}$$

for which we shall derive a bound.

In our discussion we need the following Lemma 1, directly deduced from [3].

Lemma 1. *Let f be a function defined on $[-1, 1]$. Let $\varphi^{(n)}$ be the cubic spline interpolating f at the points $\{x_i\}_{i=0}^{n+1}$ chosen as in (7), with "not-a-knot" end condition and $n \geqslant 4$.*

If $f \in C^j[-1, 1]$, $j = 1, 2$ or 3, then there exists a positive constant C, which does not depend on the mesh of nodes and on the function f, such that

$$\max_{x \in [-1, 1]} |\mathrm{D}^s(f - \varphi^{(n)})| \leqslant C h_n^{j-s} \omega(f^{(j)}, h_n), \quad s = 0, 1, \tag{11}$$

where $\mathrm{D}^s(g)$ denotes the continuous derivative of g, of order s, $h_n = \max_{0 \leqslant k \leqslant n}\{x_{k+1} - x_k\}$ and $\omega(g, \varepsilon)$ denotes the usual uniform norm modulus of continuity of g on $[-1, 1]$.

Theorem 2. *Let h_n, defined as in Lemma 1, satisfy*

$$\lim_{n\to\infty} h_n = 0 \tag{12}$$

and assume the nodes $\{x_i\}_{i=0}^{n+1}$ are chosen as in (7).

If $f \in C^j[-1, 1]$, $j = 1, 2$ or 3, then

$$R_n(uf; \lambda) = O\left(h_n^{j-1}\omega\left(f^{(j)}, h_n\right)\right), \quad n \to \infty. \tag{13}$$

Proof. From (10), if we set $r_n(x) = f(x) - \varphi^{(n)}(x)$, we can write

$$R_n(uf; \lambda) = \int_{-1}^{1} u(x)\frac{r_n(x) - r_n(\lambda)}{x - \lambda}\, \mathrm{d}x + r_n(\lambda) \,⨍_{-1}^{1} \frac{u(x)}{x - \lambda}\, \mathrm{d}x. \tag{14}$$

With the help of the mean value theorem and Lemma 1, from (14) we obtain

$$|R_n(uf; \lambda)| \leqslant C\left\{h_n^{j-1}\omega\left(f^{(j)}, h_n\right)\int_{-1}^{1} u(x)\, \mathrm{d}x + h_n^{j}\omega\left(f^{(j)}, h_n\right)\left|⨍_{-1}^{1} \frac{u(x)}{x-\lambda}\, \mathrm{d}x\right|\right\}. \tag{15}$$

From (15) the result (13) follows. □

3. Computational procedure

We first choose the set of knots $\{t_j\}_{j=-2}^{n+3}$, as described in Section 2. Moreover we assume the abscissas of integration $\{x_i\}_{i=0}^{n+1}$ as in (7).

Following [1], we can define on the set of knots

$$B_{n,j}^{(1)}(x) = \begin{cases} \left(t_j - t_{j-1}\right)^{-1}, & t_{j-1} < x \leqslant t_j, \\ 0, & \text{otherwise.} \end{cases} \tag{16}$$

The B-splines of order m (degree $m-1$) are generated by the iterative stable method [12]

$$B_{n,j}^{(m)}(x) = \frac{\left(x - t_{j-m}\right)B_{n,j-1}^{(m-1)}(x) + \left(t_j - x\right)B_{n,j}^{(m-1)}(x)}{t_j - t_{j-m}}. \tag{17}$$

We recall that the index on the cubic B-splines $\{B_{n,j}^{(4)}\}$ runs from $j = 2, \ldots, n+3$.

Therefore, if we set $\Phi_j(x) = B_{n,j}^{(4)}(x)$ in (3), as a result we have system (5).

Because the matrix with elements $\{B_{n,j}^{(4)}(x_i)\}$ is invertible and totally positive, system (5) can be solved by Gaussian elimination without partial pivoting [13].

Now we give a recurrence formula for the coefficients $\{I(uB_{n,j}^{(4)}; \lambda)\}$, which are defined by the integrals

$$I\left(uB_{n,j}^{(4)}; \lambda\right) = ⨍_{-1}^{1} u(x)\frac{B_{n,j}^{(4)}(x)}{x - \lambda}\, \mathrm{d}x. \tag{18}$$

Let $I_p(uB_{n,j}^{(m)}; \lambda)$ be the integrals

$$I_p\left(uB_{n,j}^{(m)}; \lambda\right) = ⨍_{-1}^{1} u(x)\frac{x^p B_{n,j}^{(m)}(x)}{x - \lambda}\, \mathrm{d}x, \quad p = 0, \ldots, m-q, \; q = 1, \ldots, m. \tag{19}$$

We insert (17) in (19) and find that

$$I_p\left(uB_{n,j}^{(m)};\ \lambda\right)=\frac{1}{t_j-t_{j-m}}\left[I_{p+1}\left(uB_{n,j-1}^{(m-1)};\ \lambda\right)-t_{j-m}I_p\left(uB_{n,j-1}^{(m-1)};\ \lambda\right)\right.$$
$$\left.+t_jI_p\left(uB_{n,j}^{(m-1)};\ \lambda\right)-I_{p+1}\left(uB_{n,j}^{(m-1)};\ \lambda\right)\right]. \tag{20}$$

The above recurrence formula, starting with the sequence of integrals

$$I_p\left(uB_{n,j}^{(1)};\ \lambda\right)=\int_{-1}^{1}u(x)\frac{x^pB_{n,j}^{(1)}(x)}{x-\lambda}\,\mathrm{d}x \tag{21}$$

is used to evaluate the terms

$$I\left(uB_{n,j}^{(4)};\ \lambda\right)=I_0\left(uB_{n,j}^{(4)};\ \lambda\right). \tag{22}$$

In order to calculate the elements (21) of the recurrence basis, given $t_1=-1$ and $t_n=+1$, we can write

$$I_p\left(uB_{n,j}^{(1)};\ \lambda\right)=\frac{1}{t_j-t_{j-1}}\int_{t_{j-1}}^{t_j}\frac{u(x)x^p}{x-\lambda}\,\mathrm{d}x$$
$$=\frac{1}{t_j-t_{j-1}}\sum_{k=0}^{p-1}\lambda^k\int_{t_{j-1}}^{t_j}u(x)x^{p-k-1}\,\mathrm{d}x+\lambda^pI_0\left(uB_{n,j}^{(1)};\ \lambda\right),\quad j=2,\ldots,n,$$
$$I_p\left(uB_{n,j}^{(1)};\ \lambda\right)=0,\quad \text{otherwise}. \tag{23}$$

We remark that for special weight functions u the integrals in (23) can be evaluated analytically.

In particular in the following cases:

(i) $u(x)=1$,
(ii) $u(x)=(1-x^2)^{-1/2}$,
(iii) $u(x)=(1-x^2)^{1/2}$,

we can easily derive [9,20] closed-form expressions for such integrals.

When closed-form expressions do not exist, similarly to [18], a numerical method must be used.

Table 1
$u(x)=1,\ f(x)=\mathrm{e}^x$

λ	$I(uf;\ \lambda)$	$n=4$	$n=8$	$n=16$	$n=32$
		$\lvert R_4(uf;\ \lambda)\rvert$	$\lvert R_8(uf;\ \lambda)\rvert$	$\lvert R_{16}(uf;\ \lambda)\rvert$	$\lvert R_{32}(uf;\ \lambda)\rvert$
0.1	1.99903605021	$8.3\cdot10^{-4}$	$8.9\cdot10^{-6}$	$1.4\cdot10^{-6}$	$3.2\cdot10^{-8}$
0.2	1.83919436620	$7.9\cdot10^{-5}$	$5.9\cdot10^{-5}$	$1.3\cdot10^{-6}$	$7.1\cdot10^{-8}$
0.3	1.62031402436	$1.9\cdot10^{-3}$	$4.0\cdot10^{-5}$	$2.8\cdot10^{-7}$	$9.4\cdot10^{-8}$
0.4	1.32190676341	$4.2\cdot10^{-3}$	$3.8\cdot10^{-5}$	$7.8\cdot10^{-8}$	$1.6\cdot10^{-8}$
0.5	0.91378643172	$5.9\cdot10^{-3}$	$5.1\cdot10^{-5}$	$1.2\cdot10^{-6}$	$4.7\cdot10^{-8}$
0.6	0.34815871193	$6.2\cdot10^{-3}$	$1.7\cdot10^{-5}$	$5.5\cdot10^{-6}$	$1.4\cdot10^{-7}$
0.7	−0.45916359812	$4.5\cdot10^{-3}$	$3.7\cdot10^{-4}$	$3.7\cdot10^{-6}$	$2.0\cdot10^{-8}$
0.8	−1.68484010378	$5.8\cdot10^{-4}$	$5.4\cdot10^{-4}$	$1.9\cdot10^{-6}$	$4.3\cdot10^{-7}$
0.9	−3.85323498264	$4.7\cdot10^{-3}$	$5.3\cdot10^{-5}$	$3.9\cdot10^{-5}$	$1.6\cdot10^{-7}$

4. Numerical results

In this section we give some numerical results for rules (2), based on the use of the cubic B-splines and generated by the computational procedure described in the previous section.

In order to evaluate the cubic B-splines we chose the following set of simple knots

$$t_1 = -1, \quad t_n = 1, \quad t_j = -1 + 2j/(n+1), \quad j = 2, \dots, n-1,$$

so that the nodes $\{x_i\}_{i=0}^{n+1}$, defined by (7), are equally spaced. However, we remark that, in a practical application of the rule, the function f is often known only for a given set of points. In this case it may be useful to fix the abscissas of integration at the points where f is known.

The numerical examples presented in Tables 1–6 are obtained by applying the above spline product rules to the same integrals evaluated in [14,24] by quadrature rules based on global

Table 2
$u(x) = 1$, $f(x) = (1 - x^2)^{1/2}$

λ	$I(uf; \lambda)$	$n=8$	$n=16$	$n=32$	$n=64$
		$\lvert R_8(uf; \lambda)\rvert$	$\lvert R_{16}(uf; \lambda)\rvert$	$\lvert R_{32}(uf; \lambda)\rvert$	$\lvert R_{64}(uf; \lambda)\rvert$
0.1	−0.31415926536	$2.9 \cdot 10^{-3}$	$1.1 \cdot 10^{-3}$	$3.8 \cdot 10^{-4}$	$1.3 \cdot 10^{-4}$
0.2	−0.62831853072	$1.0 \cdot 10^{-2}$	$2.3 \cdot 10^{-3}$	$7.9 \cdot 10^{-4}$	$2.7 \cdot 10^{-4}$
0.3	−0.94247779608	$1.6 \cdot 10^{-2}$	$3.6 \cdot 10^{-3}$	$1.2 \cdot 10^{-3}$	$4.3 \cdot 10^{-4}$
0.4	−1.25663706144	$1.2 \cdot 10^{-2}$	$5.6 \cdot 10^{-3}$	$1.8 \cdot 10^{-3}$	$6.3 \cdot 10^{-4}$
0.5	−1.57079632679	$1.9 \cdot 10^{-3}$	$6.4 \cdot 10^{-3}$	$2.5 \cdot 10^{-3}$	$8.8 \cdot 10^{-4}$
0.6	−1.88495559215	$2.3 \cdot 10^{-2}$	$1.3 \cdot 10^{-2}$	$3.6 \cdot 10^{-3}$	$1.2 \cdot 10^{-3}$
0.7	−2.19911485751	$9.8 \cdot 10^{-2}$	$1.3 \cdot 10^{-2}$	$5.5 \cdot 10^{-3}$	$1.8 \cdot 10^{-3}$
0.8	−2.51327412287	$1.7 \cdot 10^{-1}$	$2.0 \cdot 10^{-2}$	$1.0 \cdot 10^{-2}$	$3.0 \cdot 10^{-3}$
0.9	−2.82743338823	$1.2 \cdot 10^{-1}$	$1.2 \cdot 10^{-1}$	$1.5 \cdot 10^{-2}$	$7.8 \cdot 10^{-3}$

Table 3
$u(x) = 1$, $f(x) = (y^2 - x^2)^{-1/2}$, $y = 5$

λ	$I(uf; \lambda)$	$n=4$	$n=6$	$n=14$	$n=31$
		$\lvert R_4(uf; \lambda)\rvert$	$\lvert R_6(uf; \lambda)\rvert$	$\lvert R_{14}(uf; \lambda)\rvert$	$\lvert R_{31}(uf; \lambda)\rvert$
0.25	−0.1002688603	$1.1 \cdot 10^{-6}$	$4.5 \cdot 10^{-8}$	$1.1 \cdot 10^{-9}$	$7.5 \cdot 10^{-11}$
0.99	−1.0717993352	$1.3 \cdot 10^{-5}$	$3.1 \cdot 10^{-6}$	$1.1 \cdot 10^{-7}$	$4.1 \cdot 10^{-9}$

Table 4
$u(x) = 1$, $f(x) = (y^2 - x^2)^{-1/2}$, $y = 1.1$

λ	$I(uf; \lambda)$	$n=6$	$n=14$	$n=18$	$n=31$
		$\lvert R_6(uf; \lambda)\rvert$	$\lvert R_{14}(uf; \lambda)\rvert$	$\lvert R_{18}(uf; \lambda)\rvert$	$\lvert R_{31}(uf; \lambda)\rvert$
0.25	−0.2004430623	$1.7 \cdot 10^{-3}$	$1.2 \cdot 10^{-3}$	$7.8 \cdot 10^{-4}$	$1.2 \cdot 10^{-4}$
0.99	−7.4832176878	$4.5 \cdot 10^{-1}$	$1.1 \cdot 10^{-1}$	$6.8 \cdot 10^{-2}$	$1.6 \cdot 10^{-2}$

Table 5
$u(x)=(1-x^2)^{-1/2}$, $f(x)=(x^2+y^2)^{-1/2}$, $y=5$

λ	$I(uf;\lambda)$	$n=6$	$n=14$	$n=18$	$n=31$
		$\lvert R_6(uf;\lambda)\rvert$	$\lvert R_{14}(uf;\lambda)\rvert$	$\lvert(R_{18}(uf;\lambda)\rvert$	$\lvert R_{31}(uf;\lambda)\rvert$
0.25	−0.0012291611	$1.6\cdot10^{-7}$	$4.9\cdot10^{-8}$	$4.5\cdot10^{-8}$	$4.3\cdot10^{-8}$
0.99	−0.0046955619	$2.8\cdot10^{-6}$	$7.9\cdot10^{-8}$	$1.4\cdot10^{-7}$	$1.2\cdot10^{-7}$

Table 6
$u(x)=(1-x^2)^{-1/2}$, $f(x)=(x^2+y^2)^{-1/2}$, $y=0.1$

λ	$I(uf;\lambda)$	$n=6$	$n=14$	$n=18$	$n=24$
		$\lvert R_6(uf;\lambda)\rvert$	$\lvert R_{14}(uf;\lambda)\rvert$	$R_{18}(uf;\lambda)\rvert$	$\lvert R_{24}(uf;\lambda)\rvert$
0.25	−107.79315611	99.43	3.67	$5.9\cdot10^{-1}$	$8.1\cdot10^{-2}$
0.99	−31.256858013	104.77	1.03	$2.3\cdot10^{-1}$	$3.7\cdot10^{-2}$

polynomial interpolation at zeros of orthogonal polynomials. We remark that the results obtained appear comparable.

The absolute errors (10) are reported for different values of λ and increasing values of n.

5. Final remarks

In this paper we have presented a procedure for generating a class of product quadrature rules for (1), based on cubic spline interpolation of the function f and we have derived an error bound when $f\in C^j[-1,1]$, $j=1$, 2 or 3.

We remark that an interesting open question concerning these formulas is their convergence for a larger class of functions f, and we think that a knowledge of the behaviour of the spline $\varphi^{(n)}$ would lead to more general convergence properties for the corresponding quadrature rules.

We are investigating this problem and we propose to report our results in a forthcoming paper.

References

[1] A. Alaylioglu, D.S. Lubinsky and D. Eyre, Product integration of logarithmic singular integrands based on cubic splines, *J. Comput. Appl. Math.* **11** (3) (1984) 353–366.

[2] K.E. Atkinson, The numerical evaluation of singular integrals of Cauchy type, *SIAM J. Numer. Anal.* **9** (1972) 284–299.

[3] R.K. Beatson, On the convergence of some cubic spline interpolation schemes, *SIAM J. Numer. Anal.* **23** (1986) 903–912.

[4] M.M. Chawla and S. Kumar, Convergence of quadratures for Cauchy principal value integrals, *Computing* **23** (1979) 67–72.

[5] G. Criscuolo and G. Mastroianni, On the convergence of an interpolatory product rule for evaluating Cauchy principal value integrals, *Math. Comp.* **48** (1987) 725–735.

[6] G. Criscuolo and G. Mastroianni, On the convergence of product formulas for the numerical evaluation of derivatives of Cauchy principal value integrals, *SIAM J. Numer. Anal.* **25** (1988) 713–727.
[7] C. Dagnino, Product integration of singular integrands based on cubic spline interpolation at equally spaced nodes, *Numer. Math.* **57** (1990) 97–140.
[8] C. Dagnino and A.M. Palamara Orsi, Product integration of piecewise continuous integrands based on cubic spline interpolation at equally spaced nodes, *Numer. Math.* **52** (1988) 459–466.
[9] C. Dagnino and E. Santi, An algorithm for the generation of spline product rules for Cauchy singular integrals, Internal Report, Fac. Ingegneria, Univ. L'Aquila, 1988.
[10] P.J. Davis and P. Rabinowitz, *Methods of Numerical Integration* (Academic Press, New York, 2nd ed., 1984).
[11] C. De Boor, *A Practical Guide to Splines* (Springer, New York, 1972).
[12] C. De Boor, On calculating with B-splines, *J. Approx. Theory* **6** (1972) 50–62.
[13] C. De Boor and A. Pinkus, Backward error analysis for totally positive linear systems, *Numer. Math.* **27** (1977) 485–490.
[14] D. Elliott and D.F. Paget, An algorithm for the numerical evaluation of certain Cauchy principal value integrals, *Numer. Math.* **19** (1972) 373–385.
[15] D. Elliott and D.F. Paget, On the convergence of a quadrature rule for evaluating certain Cauchy principal value integrals, *Numer. Math.* **23** (1975) 311–319; Addendum: *Numer. Math.* **25** (1976) 287–289.
[16] D. Elliott and D.F. Paget, Gauss type quadrature rules for Cauchy principal value integrals, *Math. Comp.* **33** (1979) 301–309.
[17] A. Gerasoulis, The use of piecewise quadratic polynomials for the solution of singular integral equations of Cauchy type, *Comput. Math. Appl.* **8** (1982) 15–22.
[18] A. Gerasoulis, Piecewise-polynomial quadratures for Cauchy singular integrals, *SIAM J. Numer. Anal.* **23** (1986) 891–902.
[19] T.N.E. Greville, *Theory and Applications of Spline Functions* (Academic Press, New York, 1969).
[20] W. Gröbner and N. Hofreiter, *Integraltafel* (Springer, Wien, 1973).
[21] N.I. Ioakimidis, On the uniform convergence of Gaussian quadrature rules for Cauchy principal value integrals and their derivatives, *Math. Comp.* **44** (1985) 191–198.
[22] D.S. Lubinsky and P. Rabinowitz, Non-interpolatory integration rules for Cauchy principal value integrals, *Math. Comp.*, to appear.
[23] G.R. Miller and L.M. Keer, A numerical technique for the solution of singular integral equations of second type, *Quart. Appl. Math.* **42** (1985) 455–465.
[24] G. Monegato, The numerical evaluation of one-dimensional Cauchy principal value integrals, *Computing* **29** (1982) 337–354.
[25] R. Piessens, M. Van Roy Branders and I. Mertens, The automatic evaluation of Cauchy principal value integrals, *Angew. Informatik* **1** (1976) 31–35.